3T

УДК 004.475:[004.652:004.655]

С.С. Танянский
Харьковский национальный университет радиоэлектроники, Украина
Украина, 61166, г. Харьков, пр. Ленина, 14

Логическая модель описания и обработки данных

S.S. Tanyansky
Kharkov national university of radioelectroniks, Ukraine
Ukraine, 61166, c. Kharkov, Lenin av., 14

Logical model descriptions and data processing

С.С. Танянський
Харківський національний університет радіоелектроніки, Україна
Україна, 61166, м. Харків, пр. Леніна, 14

Логічна модель опису та обробки даних

В статье рассматриваются вопросы представления данных средствами логического программирования в задачах поддержки принятия решений. Используя синтаксис предикатов первого порядка, построена модифицированная модель описания данных за счет сокращения кванторов существования в предикатном выражении. Что позволяет группировать отрицательную информацию для обеспечения редукции логических правил к синтаксису запроса.
Ключевые слова: база данных, модель, предикат, логическое правило, квантор.

The article devoted the presentation of data by means of logic programming in problems of decision support systems. Using the syntax of first-order predicate, construct a modified model for the description of data by reducing the existential quantifiers in the predicate expression. That allows to group negative information to ensure the reduction of logical rules to the syntax of the query.
Key words: database, model, predicate logic rule quantifier.

У статті розглядаються питання представлення даних засобами логічного програмування в задачах підтримки прийняття рішень. Використовуючи синтаксис предикатів першого порядку, побудована модифікована модель опису даних за рахунок скорочення кванторів існування в предикатних виразах. Що дозволяє групувати від'ємну інформацію для забезпечення редукції логічних правил до синтаксису запиту.
Ключові слова: база даних, модель, предикат, логічне правило, квантор.

Введение

За длительную историю своего развития искусственный интеллект породил множество научных направлений, объединенных целью создания автоматизированных систем, выполняющих различные интеллектуальные функции. Одной из таких систем являются дедуктивные базы данных (ДБД). Дедуктивные базы данных основываются на языках логического программирования, что позволяет говорить об унифицированном подходе в описании и манипулировании данными на основе процедур вывода.

Основной компонентой в теории ДБД является язык представления знаний. Сближая формальный язык ДБД с естественным языком и используя механизм дедуктивного вывода, фактически получаем средство для решения более широкого круга задач, чем решение традиционными алгебраическими методами, так как в ДБД задача не решается, а описывается. В связи с этим языковые модели представления знаний в ДБД постоянно совершенствуются, позволяя решать все более широкий класс задач. С другой стороны, высокий уровень абстракции, поддерживаемый язы-

ковой моделью ДБД, требует развитых средств интерпретации программы ДБД как системы взаимосвязанных методов и моделей анализа и синтеза плана вычисления результата запроса.

Вначале 70-х гг. начали формироваться подходы в технологиях баз данных (БД), связанные с использованием в качестве модели данных исчисление предикатов. Работы в этой области велись в различных направлениях, одно из которых является началом создания систем ДБД. Развитие теории ДБД определило активное исследование проблем организации данных и описания предметных областей, которое проводилось многими учеными (А.А. Стогний [1], Н.И. Буслик [2], Ю.П. Шабанов-Кушнаренко [3], Д.А. Поспелов [4], А.В. Замулин [5], П. Джексон [6] и др.).

Особый вклад в теорию ДБД внес Дж. Ульман, давший описание логического языка ДБД – DataLog [7]. Теоретические исследования в этой области выполнены М.И. Дехтярем и А.Я. Диковским [8]. Также опубликованы монографии и статьи, например А. Фернандеса и Дж. Минкера [9], С. Чери [10] и др.

В связи с актуальностью рассматриваемой проблемы, а также с необходимостью расширения возможностей обрабатывать слабоструктурированные данные в информационных системах поддержки принятия решений целью статьи является построение логико-предикативных моделей на основе ДБД.

Элементы логико-предикативной модели. Задача декларативного решения задачи сводится к определению истинности логической формулы – программы ДБД, соответствующей описанию предметной области (ПрО).

Существуют два подхода к определению истинностного значения логических формул: теоретико-модельный и теоретико-доказательный. В теории моделей логическое предположение истинно, если оно истинно во всех допустимых интерпретациях. В теории доказательств предположение истинно, если оно может быть выведено из некоторого набора аксиом с помощью правил вывода [10], [11].

Рассмотрим основные элементы логико-предикативной модели. Пусть $D = \{D_1,..,D_n\}$ – множество сущностей некоторой ПрО. Каждому предикату $p(X_1,...,X_n)$ сопоставим отношение $R(Y_1,...,Y_n)$, где $R \in D$ и X_i, Y_i – предикатные переменные. Таким образом, используя пропозициональные связки и кванторы всеобщности и существования, определяются произвольные формулы в D.

Если значение логической формулы f для D является истинным, то D является моделью f. Если D является моделью для каждой из формул множества F, то D является моделью множества F. Если во всех моделях F формула f принимает значение «истинно» и при этом $f \notin F$, то f является логическим заключением F, в обозначении $F \models\!\Rightarrow f$.

В [12] Эрбраном предложено ограничиться рассмотрением интерпретаций определенного типа без потери их общности. Интерпретации Эрбрана дают символам констант одни и те же значения. Каждый n-местный предикатный символ интерпретируется на двухэлементное множество {истинно, ложно}. Для того чтобы определить интерпретацию Эрбрана $\Im$ для некоторой программы, необходимо задать истинностные значения основным фактам. Очевидно, что всегда выполняется вхождение $\Im \subseteq HB$, где HB – базис Эрбрана.

Определим истинностное значение фактов и правил DataLog:

1. Если g – некоторый произвольный факт, то g принимает значение «истинно» в случаях: а) $g = True|(g) \wedge g \in \Im$, или б) $g = True|(\neg g) \wedge |g| \notin \Im$, где $|g|$ – дополнение g. В противном случае g принимает значение «ложно».

2. Пусть L – правило вида l_0 :- $l_1,..., l_n$, где среди l_i; ($i = \overline{1,n}$) допускаются отрицательные элементы (литералы). Значение L в $\Im$ является «истинным», если $\forall\ \Theta(L)\ \ \exists\ l_i\Theta \in \Im \mid l_0\Theta \in \Im$ ($i = \overline{1,n}$), где Θ – подстановка, которая определяется как конечное множество вида $\{X_1/t_1,..., X_n/t_n\}$ и $X_i \neq t_i$. Если l – литерал, то $l\Theta$ обозначает литерал, где каждый терм t_i заменяется на $t_i\Theta$. Например, $l = p(a,X,Y,d)$, $\Theta = \{X/c, Y/b\}$, $l\Theta = p(a,c,b,d)$.

Очевидно, что логическая программа имеет множество моделей, однако в общем случае модель может включать в качестве собственных подмножеств другие модели. Например, для некоторой программы S:

S : $p(a)$.

$q(X)$:- $p(X)$.

интерпретации $\Im_1 = \{p(a),\ q(a)\}$, $\Im_2 = \{p(a),q(a),q(b)\}$ являются моделями Эрбрана программы S.

Очевидно, что наибольший интерес представляют минимальные модели, то есть те модели, в которых отсутствуют факты, не влияющие на истинностные значения фактов и правил программы. В рассмотренном примере минимальной моделью является $\Im_1$.

Как отмечено в [12], DataLog не предоставляет эффективного алгоритма поиска минимальной модели. Для решения практических задач в основном используется теоретико-доказательный подход к определению истинности логических формул. В отличие от теоретико-модельного подхода такой подход позволяет определять истинность логической формулы, не прибегая к интерпретации и модели, используя только синтаксические преобразования логических формул.

На практике могут быть случаи, когда минимальная модель может содержать бесконечное множество фактов. Для получения корректного результата необходимо исключать ситуации вывода бесконечных моделей. Для этого рассмотрим условия безопасности на вид дизъюнктов DataLog, которые состоят в следующем:

– каждая переменная головы правила должна присутствовать в теле этого правила;
– имя головы правила должно принадлежать множеству имен интенсиональных (вычисляемых) предикатов.

Расширение предикатной модели. Обозначим предикатную модель парой: $<p^{\omega}, \Im_{p^{\omega}}>$, где p^{ω} – множество дизъюнктов, $\Im_{p^{\omega}}$ – модель p^{ω}, ω – версия DataLog.

Так как переменные дизъюнктов DataLog связываются квантором всеобщности, это приводит к определенным сложностям при записи выражений, содержащих квантор существования [12]. Обобщенная форма таких выражений имеет вид:

$$\forall X_1,\ldots,X_n \wedge p_1(Y_{11},\ldots,Y_{1m})\ \delta_1(\exists Z_1,\ldots,Z_k \wedge p_2(Y_{21},\ldots,Y_{2m}) \wedge \delta_2(\exists Z_{k+1},\ldots,Z_p \wedge \ldots)\ldots), \quad (1)$$

где δ_i – либо пустой символ, либо символ отрицания.

Выражение, следующее за символом δ_i, называют кванторным выражением. Переменные кванторного выражения называют внешними, если они входят в конъюнкции слева от него, остальные переменные назовем внутренними.

Исключим в выражении (1) кванторы существования и заменим конъюнкцию запятой, добавим полученную конструкцию в синтаксис DataLog, расширим его мощность, с одной стороны, возможностью выражать отрицательную информацию, с

другой – дополнительной возможностью группировать информацию с помощью кванторного выражения.

Пусть правило DataLog имеет следующий вида:

$$L_0 :- L_1, L_2,..., L_n, \rho, \quad (2)$$

где $L_1, L_2,..., L_n$ – положительные литералы, ρ – кванторное выражение.

Значение правила (2) в интерпретации $\Im$ принимает значение «истинно», если справедливо выражение:

$$\forall t \,|\, L_1 t \in \Im,..., L_n t \in \Im \; \exists t' \; (t \cap t' = \varnothing) \,|\, \forall L_i \in \rho \,|\, L_i t t' \in \Im \wedge L_0 t \in \Im \quad (3)$$

Обобщение определения истинностного значения на случай, когда правило DataLog имеет более одного кванторного выражения, очевидно.

Рассмотрим метод поиска истинностного значения программ DataLog. Пусть кванторное выражение в исчислении логики предикатов первого порядка имеет вид:

$$\forall X_1,\ldots,X_n \; \exists Y_1,\ldots,Y_n \wedge L_m(\ldots,Z_k,\ldots), \quad (4)$$

где $Z_k \in \{X_1,\ldots,X_n,Y_1,\ldots,Y_n\}$.

Применяя правило введения сколемовских констант [11] n раз, можно исключиючить в кванторном выражении все кванторы существования, таким образом получим выражение:

$$\forall X_1,\ldots,X_n \wedge L_m(\ldots,Z_k,\alpha_i\ldots), \quad (5)$$

где $Z_k \in \{X_1,\ldots,X_n\}$, α_i – предметные константы ($i = \overline{1,n}$).

Константы α_i можно заменить свободной переменной с уникальным именем, так как кванторное выражение принимает истинностное значение только в интерпретациях, содержащих эти константы. Таким образом, выражение (5) примет вид:

$$\forall X_1,\ldots,X_n \wedge L_m(\ldots,Z_k,A_i\ldots), \quad (6)$$

где A_i – свободная переменная ($i = \overline{1,n}$).

Введем k-арный предикат p, в котором область определения совпадает с доменом полученной предикатной формулы:

$$p(X_1,\ldots,X_n) :- L_m(\ldots,Z_k,\ldots),..., L_{m+n}(..., Z_s,...). \quad (7)$$

Так как область значений предиката совпадает с областью значений, исходное правило можно привести к виду без кванторных выражений:

$$L_0 :- L_1, L_2,..., L_n, p(X_1,\ldots,X_n). \quad (8)$$

Если предикат содержит отрицательное кванторное выражение, то выражение (8) примет вид:

$$L_0 :- L_1, L_2,..., L_n, \neg p(X_1,\ldots,X_n). \quad (9)$$

Отметим, что ни одна переменная из множества $\{X_1,\ldots,X_n\}$ не встречается в голове правила. Таким образом, правила (8) и (9) можно редуцировать в правила чистого DataLog. Если правило имеет составное кванторное выражение, то для каждого входящего выражения строится соответствующее правило.

С учетом проведенного обобщения можно утверждать, что любое правило расширенной модели может быть представлено совокупностью правил чистого DataLog.

Таким образом, сохраняя нумерацию условий безопасности, можно сформулировать третье условие безопасности для расширенной модели DataLog: кванторное выражение должно включать, по крайней мере, одну внешнюю переменную.

Выводы

Рассмотренный в статье материал показал, что язык логического программмирования как предикатная модель представления знаний является одним из самых мощных и универсальных средств описания ПрО. Кроме того, теоретико-доказательный подход определяет средства решения задач исключительно на основе описания ее ПрО, правил порождения и выбора альтернатив, поэтому класс решаемых задач определяется дескриптивной мощностью модели представления знаний.

Для создания единой предикатной модели, выступающей в качестве основы языковой модели представления данных, рассмотрено расширение модели с полными кванторными выражениями, позволяющее группировать отрицательную информацию для обеспечения редукции правил к синтаксису чистого DataLog.

Фундамент предикатных моделей образует хорошо изученное исчисление предикатов. Однако практическая реализация отдельных положений исчисления предикатов вызывает трудности, что определяет актуальность дальнейшего развития выразительных возможностей предикатных моделей.

Несмотря на формальное обоснование семантики расширенной модели логиического языка, практическая реализация в среде систем управления базами данных невозможна без дополнительных средств анализа и интерпретации его программ. Таким образом, дальнейшие исследования в данном направлении должны быть направлены на разработку средств поддержки логических языков описания и обработки данных.

Литература

1. Стогний А.А. Средства и методы автоматизированного проектирования реляционных баз данных / А.А. Стогний // Материалы III Всесоюз. конф. «Банки данных». – Таллин. – 1985. – С. 103-115.
2. Буслік М.М. Оптимальні зображення реляційних баз даних. / М.М. Буслік // Монографія. – К.: ІСДО, 1993. – 84 с.
3. Шабанов-Кушнаренко Ю.П. Теория интеллекта: Проблемы и перспективы / Ю.П. Шабанов-Кушнаренко. – Харьков: Вища шк. Изд-во при ХГУ, 1987. – 158 с.
4. Поспелов Д.А. Логико-лингвистические модели в системах управления / Д.А. Поспелов – М.: Энергия, 1981.-231 с.
5. Замулин А.В. Системы программирования баз данных и знаний / А.В. Замулин. – Новосибирск: Наука, 1990. – 352 с.
6. Джексон П. Введение в экспертные системы / П. Джексон. – М.: "Вильямс", 2001. – 624 с.
7. Ullman J.D. Principles of Database and Knowledge-Base Systems / J.D. Ullman – Computer Science Press, 1988 – 1137 p.
8. Дехтярь М.И. Динамические дедуктивные базы данных / М.И. Дехтярь, А.Я. Диковский // Техническая кибернетика. – 1994. – №5. – С. 55-67.
9. Gallaire H. Logic and Databases: A Deductive Approach / H. Gallaire, J. Minker, J. Nikolas // ACM Computing Surveys. – 1984. – № 1. – P. 154-185.
10. Чери С. Логическое программирование и базы данных / С. Чери, Г. Готлоб, Л. Танка. – М.: Мир, 1992.-352 с.
11. Мендельсон Э. Введение в математическую логику / Э. Мендельсон – М.: Наука, 1984. – 320 с.
12. Осуга С. Обработка знаний / С. Осуга. – М.: Мир, 1989. – 293 с.

RESUME

S.S. Tanyansky

Logical Model Descriptions and Data Processing

The article devoted the structural component of a relational data structure describing the means of deductive databases.

On the basis of the results proposed in [2], [3] the necessity to use logic programming for knowledge processing tasks in support of decision-making. It is shown that the use of structural component models of predicate logic and meets the necessary requirements of the conception and allows implementing all the required functionality.

In view of the structural features of the logical model of knowledge representation, a modification of the predicate forms describes the data. The extended model allows reducing the predicate expression and improving the efficiency of the reduction of logical rules to the syntax of the query. On the basis of the considered transforms a method for searching the truth value of a logic program which can significantly simplify the design rules of inference result.

The implementation of data management system with the proposed model solves the problem of knowledge processing within a reasonable time considering the possible structural heterogeneity of information accumulated through the using of intelligent data presentation.

Статья поступила в редакцию 27.02.2013.